

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application.

Listing of Claims:

1 (currently amended): A method for releasing a product under development, the method comprising:

storing in memory ~~an~~ a first association between a first version (~~hereinafter "current version"~~) of a component of the product and a first time (~~hereinafter "future tick"~~) selected from periodically recurring times in future; and

storing in memory a second association between a second version of the component and a second time selected from periodically recurring times in future; and

~~identifying a version (hereinafter "release version") of the component as being available for release, using the association and a time (hereinafter "most recent tick") that occurred most recently in past, from among periodically recurring times~~

identifying one of said versions as a release version, by using said associations and depending on whichever one of said times occurred most recently in the past.

2 (currently amended): The method of Claim 1 wherein:

the release version is different from ~~the current~~ the second version if ~~the future tick~~ if the second time is yet to occur.

3 (currently amended): The method of Claim 1 further comprising:

the release version is the ~~current~~ first version if ~~the future tick is the most recent tick~~ first time occurred most recently in the past, from among periodically recurring times.

4 (currently amended): The method of Claim 1 further comprising:

on passage of said ~~future tick (hereinafter "past tick")~~, first time, storing in memory another association, of said first version with a third time in future (~~hereinafter "next tick"~~)

that is scheduled to occur immediately after said ~~past tick~~, first time, among the periodically recurring times.

5 (original): The method of Claim 1 wherein the association is stored in a record of a database.

6 (currently amended): The method of Claim 1 further comprising:
storing an identity of a person responsible for development of ~~the current~~ each version.

7 (currently amended): The method of Claim 1 wherein the component comprises software, the method further comprising:
storing an address of ~~the current~~ each version.

8 (original): The method of Claim 1 wherein the component comprises software, the method further comprising:
copying the software to a central location of storage of other components.

9 (currently amended): The method of Claim 1 further comprising:
storing an identity of a bug that has been fixed in ~~the current~~ each version.

10 (original): The method of Claim 1 further comprising:
storing a label of the component used in a version control system.

11 (currently amended): The method of Claim 1 further comprising:
storing an indicator (~~also called "staging time"~~) of when the ~~current~~ first version is associated with the ~~future tick~~ first time.

12 (currently amended): The method of Claim 1 further comprising:
storing for the ~~current~~ first version an identity of a release of all components in which the ~~current~~ first version is to be included.

13 (currently amended): The method of Claim 12 wherein:
the release is predetermined to occur subsequent to ~~a time (hereinafter "milestone tick") selected from one of~~ the periodically recurring times selected to be a milestone time.

14 (currently amended): The method of Claim 13 wherein:
~~said future tick is one of a plurality of ticks prior to the milestone tick.~~
said first time is one of a plurality of times prior to said milestone time.

15 (currently amended): The method of Claim 1 further comprising:
receiving said ~~future tick~~ first time and an identification of said first version via a graphical user interface.

16 (currently amended): The method of Claim 1 further comprising:
storing an association of a unique identifier with the ~~current~~ first version.

17 (currently amended): The method of Claim 16 further comprising:
receiving the unique identifier from a person responsible for development of the first version.

18 (original): The method of Claim 1 wherein:
said periodically recurring times occur once a week.

19 (currently amended): The method of Claim 18 wherein:
each tick time in the periodically recurring times occurs on a predetermined day selected from a group consisting of Tuesday, Wednesday and Thursday.

20 (currently amended): The method of Claim 18 wherein:
each tick time in the periodically recurring times occurs on Wednesday.

21 (currently amended): The method of Claim 1 wherein ~~said current version is hereinafter "first version", the method further comprising:~~

~~storing in memory another association between another version (hereinafter "second version") of the component and said future tick;~~

~~wherein after the future tick~~ after the second time has occurred, said second version is identified as the release version.

22 (currently amended): The method of Claim 21 further comprising:

storing a ~~first~~ third association of a first identifier with the first version; and

storing a ~~second~~ fourth association of a second identifier with the second version.

23 (currently amended): The method of Claim 21 wherein:

said storing of ~~another~~ second association is performed prior to said ~~future tick~~ second time.

24 (currently amended): The method of Claim 21 wherein:

said storing of ~~another~~ second association is performed subsequent to said ~~future tick~~ second time only as an exception.

Claim 25 (canceled).

26 (original): A computer readable storage medium encoded with software instructions to perform the method of Claim 1 when executed by a computer.

27 (original): A signal embedded in a carrier medium and encoded with software instructions to perform the method of Claim 1 when executed by a computer.

28 (currently amended): A computer programmed to track each component of a product under development, the computer comprising:

memory holding a first version of a component and a first time (~~hereinafter "tick"~~) that is one of several periodically recurring times in future, at memory locations addressed by a data structure;

first means for determining from current time, a second time (~~hereinafter "most recent tick"~~) that occurred most recently in the past, from among periodically recurring times; and

second means coupled to the first means to receive therefrom the ~~most recent tick~~ second time and coupled to the memory to receive therefrom ~~one or more pairs (of version and tick) data~~ data addressed by the data structure, the second means identifying a ~~version (hereinafter "release version")~~ the first version as being available for release if the tick ~~paired to the release version~~ first time matches the ~~most recent tick~~ second time.

29 (currently amended): The computer programmed as in Claim 28 wherein:

said memory also holds an identity of a person responsible for development of said first version, an address of the first version, an identity of a bug that has been fixed in the first version, a label of the component used in a version control system for the first version, an indicator of when the first version is associated with the tick, and an identity of a release of all components in which the first version is to be included.